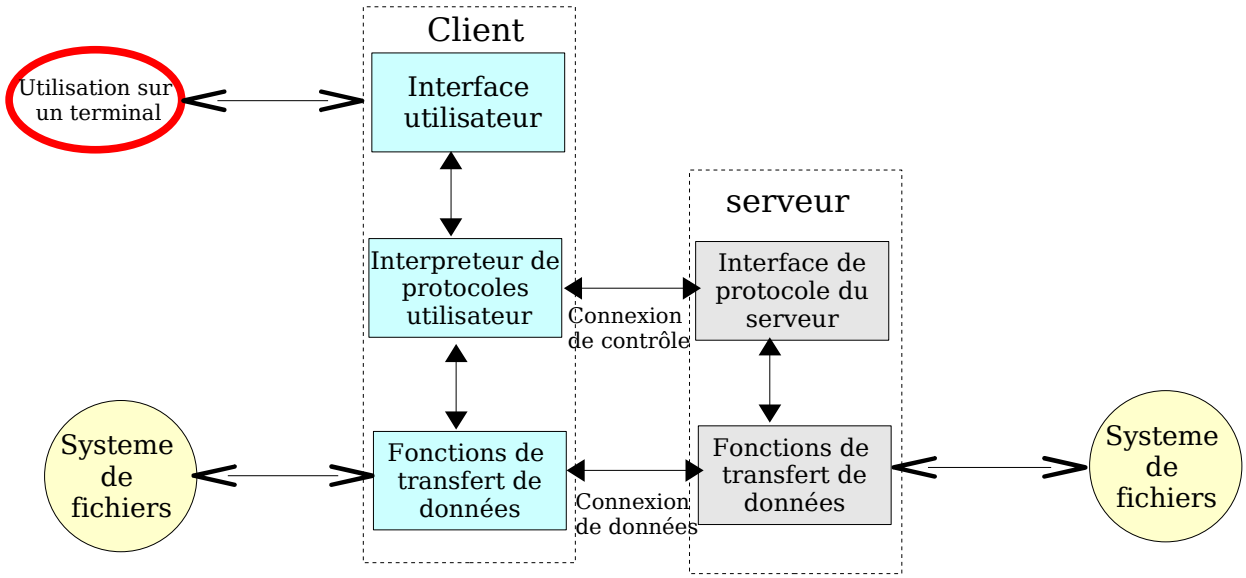
FTP

FTP est défini au-dessus de TCP et utilisent deux connexions TCP IP pour fonctionner comme illustré dans la figure suivante



Tansfert de fichier par FTP.

Tout d'abord on y voit que le client utilise FTP à travers une interface qui peut être graphique (logiciels gFTP, WS-FTP, Fetch, ...) ou texte (mode commandes d'unix par exemple).

La *connexion de contrôle* est établie de façon normale en mode client serveur sur le **port 21** du serveur **et sur un port aléatoire du client** pour tout ce qui est de type transfert interactif.

Elle sert donc *tout le temps de la session* à transférer les commandes du client et presque toutes les réponses du serveur.

La *connexion de données* sert à transférer les fichiers et les contenus de répertoires du serveur, c'est-à-dire tous les transferts de masse.

En effet, lorsque le client demande le contenu d'un répertoire la réponse peut être très longue et il est préférable de l'envoyer sur cette connexion plutôt que sur celle du transfert interactif. À chaque fois qu'un fichier doit être transféré, dans un sens ou dans l'autre, le client initie une connexion de données en s'attribuant un port et envoie au serveur une demande de connexion sur la connexion de contrôle. Le serveur se sert du numéro de port reçu pour établir la connexion de données entre son **port 20** et ce port indiqué par le client.

Client FTP en ligne de commande

Le client FTP est un logiciel qui va nous permettre de transférer des fichiers de notre ordinateur vers un serveur et réciproquement. Le client FTP est livré en standard sous Linux. Il est présent sous Windows pour peu que vous ayez installé le protocole TCP/IP.

ouvrons une console.En tapant

ftp alpha.poitou-charentes.iufm.fr

je demande à mon client ftp de se connecter au **serveur alpha.poitou-charentes.iufm.fr**

```
[etudiant@pc01]$ ftp alpha.poitou-charentes.iufm.fr

Connected to alpha.poitou-charentes.iufm.fr.

220 ProFTPD 1.2.5 Server (ProFTPD Default Installation)
[alpha.poitou-charentes.iufm.fr]

Name (alpha.poitou-charentesIufm.fr :bondazm) : anonymous
```

Le serveur me demande de m'identifier. Je veux me conecter en tant qu'anonyme donc anonymous

```
331 Anonymous login ok, send your complete email address as your password.
Password :
230 Anonymous access granted, restrictions apply.
```

Aie ! On a des droits réduits.

```
Remote system type is UNIX
```

Remote : machine distante, par opposition, local : votre station.

```
Using binary mode to transfer files.
```

Par défaut on utilise le mode binaire. Ça tombe bien on va dialoguer de unix à unix donc pas de problèmes.

Utilisation des FTP ANONYME :

Le FTP anonyme est un moyen simple de mettre des fichiers a disposition de tous (le plus souvent en lecture seule pour limiter le danger). Les serveurs FTP anonymes acceptent les connexions de n'importe qui, les authorisant a acceder aux fichiers publiques.

Autre méthode:

Entrez les commandes de connexion directement dans une console ftp:

```
ftp
open multimania.com
user tarzan
pass ********
```

Quelques commandes

```
ftp>
```

Ici le serveur FTP attends des ordres. Vous ne savez pas, alors tapez help

```
Ftp> help
```

Vous devez avoir tous les ordres possibles.

```
Ftp> ls
227 Entering Passive Mode (192,168,16,3,128,11).
150 Opening ASCII mode data connection for file list
drwxr-sr-x 2 root ftp 4096 Oct 14 21 :57 pub
226-Transfer complete.
```

ls *est le même que celui de la console. Il va vous afficher tous les répertoires et fichiers disponibles sur le serveur.*
Selon les réglages du serveur, vous pouvez tomber directement dans le répertoire "pub" ou sont rangés les fichiers qui vous intéressent auquel cas passez directement à l'étape 2 ou dans un répertoire racine du service ftp. Dans ce cas vous obtenez quatre répertoires :
bin, etc., lib : la tripaille système. C'est root sur le serveur qui gère son serveur FTP avec les infos contenues dans ces répertoires.
pub : le classique répertoire où sont stockés les fichiers disponibles.
Allons donc dans ce répertoire :

```
Ftp>cd pub
250 CWD command successful.
```

Tout c'est bien passé.

Etape 2 : le transfert de fichiers (get et put)

Pour télécharger un fichier distant et le stocker sous un autre nom sur votre machine locale : `get` *`distant-file local-file`*. Pour télécharger simplement un fichier, pas besoin de spécifier un nouveau nom : `get` *`disant-file`*.

Même chose pour envoyer un fichier de votre machine local vers le serveur : `put` *`local-file distant-file`*. Pour l'envoyé simplement : `put` *`local-file`*.

```
Ftp>ls
250 CWD command successful.
```

Oh joie ! Il y a pleins de fichiers.
*Il y a en particulier un mode d'em**ploi de vi qui m'intéresse.***
Ftp>get mode_demploi_de_vi
get *pour prendre, récupérer le fichier situé sur le serveur et le copier sur notre machine.*

```
ftp> get mode_demploi_de_vi
local : mode_demploi_de_vi remote : mode_demploi_de_vi
227 Entering Passive Mode (192,168,16,3,128,4).
150 Opening BINARY mode data connection for mode_demploi_de_vi (1681 bytes).
226 Transfer complete.
1681 bytes received in 0.0016 seconds (1e+03 Kbytes/s)
```

Nous venons de télécharger un fichier du serveur sur notre machine.

```
Ftp> !ls
```

le ! C'est pour indiquer la machine locale.

Remarques :
Refaites-le

ftp>get mode_demploi_de_vi
Notez qu'il n'y a aucun message d'alerte, ftp écrase gentiment tout ce qui est sur sa route.
ftp>hash
Les impatients verront un tas de # défiler sur leur écran et éviteront de s'arracher les cheveux en pensant qu'il ne se passe rien.

A votre avis : un anonyme peut'il mettre quelque chose sur le serveur ?
Ceux qui ont un doute essaieront :

```
ftp> put mode_demploi_de_vi
local : mode_demploi_de_vi remote : mode_demploi_de_vi
les deux fichiers auront le même nom. C'est heureux !

227 Entering Passive Mode (192,168,16,3,128,9).
550 mode_demploi_de_vi : Permission denied
```

En tant que root, je suis rassuré. Les hostiles ne mettront pas n'importe quoi sur mon serveur.

Une petite finesse, je veux utiliser un autre répertoire sur la machine locale :
ftp>**lcd** /michel/import
l pour local, cd : comme d'ab ! pour change directory.
Maintenant tout fichier arrivera ou partira de ce répertoire.
On en a fini avec anonymous.
Ctrl-D pour se déconnecter ou quit ou exit ou bye.
ftp> bye
221 Goodbye.
On a indiqué ici qu'un nombre minimal d'ordres possibles. C'est, le minimum à savoir. Pour les accros, help vous donne tous les ordres possibles.

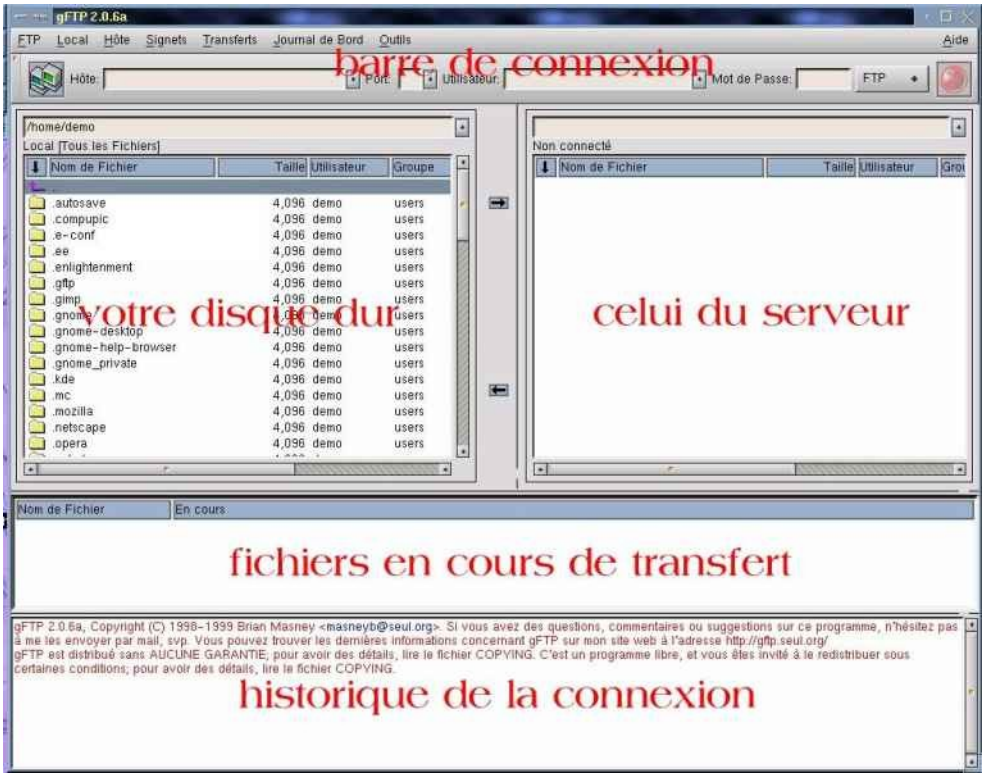
!	Ouvre un shell sur l'Operating System
?	Appelle le module d'aide
ascii	Configure le mode de transfert de fichiers en tant que ASCII
binary	Configure le mode de transfert de fichiers en tant que BINAIRE
cd	Change le répertoire courant sur le poste distant
close `disconnect`	Termine la connexion avec le poste distant
delete	Efface un fichier sur le poste distant
dir	Sort la liste des fichiers et répertoires présents dans le répertoire courrant sur le poste distant
get	Récupère un fichier depuis le poste distant
hash	Affiche # (dièse ou '*hash character*') pour chaque bloc de données transféré
lcd	Change le répertoire sur le poste local (localhost)
ls	Affiche la liste des fichiers et répertoires présents dans le répertoire courrant sur le poste distant
mdelete	Suppression de fichiers multiples sur le poste distant

mdir	Création d'un répertoire sur le poste distant
mget	Récupèration de fichiers multiples sur le poste distant mget *.wav mput *.gif ../images
mkdir	Création d'un répertoire sur le poste distant
mput	Emission de fichiers multiples sur le poste distant depuis le poste local.
open	Ouvre une connexion avec le poste distant
put	Emission d'un fichier vers le poste distant
pwd	Affiche le répertoire courrant sur le poste distant
quit `bye`	Termine la session FTP
quote	Exécute une commande sur le serveur
recv	Récupère un fichier depuis le poste distant
rmdir	Supression d'un répertoire entier
send	Emission d'un fichier vers le poste distant
type	Affiche le mode courrant de transfert de fichiers (ASCII ou BYNARY)
verbose	Charge/Décharge le mode "bavard" , commencer ou arreter les prompts en texte plein
`reget`	Equivalent de `get` mais ici le transfert est reprit à la fin du fichier, pratique pour reprendre un téléchargement précédemment interrompu.

Pour les gens qui n'aiment pas taper au clavier, il existe des clients FTP en mode graphique. Apparemment plus simples ils devraient vous faciliter la tâche. Surtout que, maintenant, vous pourrez comprendre ce qui se passe.
Essayez Client FTP graphique : gFTP

Client gFTP (g pour Gnome)

La fenêtre telle qu'elle apparaît au démarrage :


gFTP 2.0.6a
FTP Local Hôte Signets Transferts Journal de Bord Outils Aide
barre de connexion
Hôte: Port: Utilisateur: Mot de Passe: FTP
/home/demo
Local [Tous les Fichiers]
Nom de Fichier Taille Utilisateur Groupe
votre disque dur
Non connecté
Nom de Fichier Taille Utilisateur
celui du serveur
Nom de Fichier En cours
fichiers en cours de transfert
historique de la connexion

Presque le jumeau de FileZilla

Description :

gFTP est un client FTP multithread sous Linux (X-Window) écrit en GTK. Son interface est similaire à celle de *Cute FTP* sous Windows.

Points forts :

- Une interface graphique facile à prendre en main, claire et fonctionnelle
- Gestion complete des signets
- Reprise des transferts interrompus
- Possibilité de faire plusieurs transferts en même temps
- Possibilité de définir un programme pour la visualisation et l'édition des fichiers. Ce qui permet de travailler directement sur le serveur.
- Le mode passif est selectionné par défaut, pratique pour les câblés...
- Support du FXP (File eXchange Protocol) pour la version 2.0.7b :
  On se connecte à deux sites, le premier dans la vue de gauche, le second à droite. Et on transfert les fichiers de l'un à l'autre sans passer par son disque local.
- Comparaison des fenêtres :
  gFTP se base soit sur la taille des fichiers, leur noms ou leur dates de modification, selectionne alors les fichiers différents sur le disque local et le serveur. Il suffit alors de cliquer sur une des deux flèches pour downloader ou uploader les fichiers ainsi selectionnés. Pratique pour ne télécharger que ce qui a été modifié.
- Gestion du "drag & drop"

Points faibles :

- Pas de mode de selection automatique du type de transfert (ascii ou binaire).

- quelques blocages même en utilisant le mode passif sur le serveur . Ça semble ce résoudre en décochant l'option *"utiliser la mémoire cache"* dans la fenêtre de configuration.
- Pour les pocesseurs de souris à deux boutons il faut émuler le troisième pour beneficier de la gestion du "drag & drop"

Pratique des Serveurs Ftp

Configurer le serveur ProFTP (mandrake)

Nous avons installé ProFTP (par défaut sous mandrake sinon, récuperez les rpm)
vérifiez en tapant ftp localhost

```
[michel@bureau michel]$ ftp localhost
Connected to localhost.
220 ProFTPD 1.2.5 Server (ProFTPD Default Installation)
[bureau.perso]
500 AUTH not understood.
500 AUTH not understood.
KERBEROS_V4 rejected as an authentication type
Name (localhost:michel): michel
331 Password required for michel.
Password:
230 User michel logged in.
Remote system type is UNIX.
Using binary mode to transfer files.
ftp>
```

```
ftp> cd /
250 CWD command successful.
ftp> ls
227 Entering Passive Mode (127,0,0,1,128,12).
150 Opening ASCII mode data connection for file list
-rw-r--r--   1 root     root            0 Nov 15 10:18 1
drwxr-xr-x   2 root     root         4096 Nov 15 10:06 bin
drwxr-xr-x   3 root     root         4096 Jan 12 11:04 boot
drwxr-xr-x   1 root     root            0 Jan  1  1970 dev
drwxr-xr-x  60 root     root         4096 Jan 12 11:04 etc
drwxr-xr-x   4 root     root         4096 Dec  9 15:38 home
drwxr-xr-x   2 root     root         4096 Nov 15 09:36 initrd
drwxr-xr-x   9 root     root         4096 Dec 16 14:32 lib
drwxr-xr-x  12 root     root         4096 Dec 29 17:58 mnt
drwxr-xr-x   3 root     root         4096 Nov 22 18:49 opt
dr-xr-xr-x 116 root     root            0 Jan 12 12:02 proc
drwxrwxrwx   3 root     root         4096 Nov 18 14:27 public
drwx------  16 root     root         4096 Jan 10 13:45 root
drwxr-xr-x   2 root     root         4096 Dec 16 14:32 sbin
drwxrwxrwt  92 root     root         4096 Jan 12 11:07 tmp
drwxr-xr-x  12 root     root         4096 Nov 22 18:52 usr
drwxr-xr-x  20 root     root         4096 Nov 15 09:57 var
226-Transfer complete.
226 Quotas off
ftp>
ftp> bye
```

```
221 Goodbye.
```

Deux défauts 'apparaissent :
- Les "anonymes" ne peuvent pas se connecter à notre serveur
- Les clients authentifiés peuvent se balader partout

Il faut donc les régler en affinant le paramètrage.
Commencons par créer un double du fichier original que nous allons trafiquer.
Le fichier qui nous interesse est /etc/proftpd.conf

Faites en un double que vousnommerez proftpd.conf.original.
Lancer un éditeur de texte et ouvrez /etc/proftpd.conf

Les anonymes

Tout ce que nous avons à faire ici est de créer la section anonymous pour que les clients puissent se connecter en anonymous.

```
<Anonymous ~ftp>
 User ftp
 Group ftp
 UserAlias anonymous ftp
 MaxClients 10
 RequireValidShell off
 AnonRequirePassword off
 <Limit WRITE>
 DenyAll
 </Limit>
 </Anonymous>
```

L'utilisateur anonymous sera un alias du compte ftp, 10 connexions maxi, sans shell de commande, sans mot de passe valide, sans droit d'écriture.
Son répertoire:
grep ftp /etc/passwd
ftp:x:76:76:system user for proftpd:**/var/ftp**:/bin/false

Les authentifiés

On limite la capacité de navigation des utilisateurs authentifiés à leur seul répertoire. Le réglage par défaut leur permet en effet d'aller se ballader partout dans l'arborescence.

C'est à la dernière ligne : **DefaultRoot ~**

Le ~ indique le répertoire utilisateur.

On dit que l'utilisateur a été "**chrooté**" (changement de racine)
chroot remplace le répertoire racine du processus en cours par celui spécifié par le chemin . Ce répertoire sera utilisé comme origine des chemins commençant par /. Le répertoire racine est hérité

par tous les enfants du pro cessus ayant fait le changement.

Vous devez avoir un fichier proftpd.conf qui ressemble à ça :

```
# This is a basic ProFTPD configuration file (rename it to
 # 'proftpd.conf' for actual use.  It establishes a single server
 # and a single anonymous login.  It assumes that you have a user/group
 # "nobody" and "ftp" for normal operation and anon.
 ServerName        "ProFTPD Sur linux"
 ServerType        standalone
 DefaultServer     on
 # Allow FTP resuming.
 # Remember to set to off if you have an incoming ftp for upload.
 AllowStoreRestart          on
 # Port 21 is the standard FTP port.
 Port    21
 # Umask 022 is a good standard umask to prevent new dirs and files
 # from being group and world writable.
 Umask    022
 # To prevent DoS attacks, set the maximum number of child processes
 # to 30.  If you need to allow more than 30 concurrent connections
 # at once, simply increase this value.  Note that this ONLY works
 # in standalone mode, in inetd mode you should use an inetd server
 # that allows you to limit maximum number of processes per service
 # (such as xinetd)
 MaxInstances      30
 # Set the user and group that the server normally runs at.
 User    nobody
 Group nogroup
 # Normally, we want files to be overwriteable.
 <Directory /*>
  AllowOverwrite on
 </Directory>
 # Needed for NIS.
 PersistentPasswd  off
 <Anonymous ~ftp>
 User ftp
 Group ftp
 UserAlias anonymous ftp
 MaxClients 10
 RequireValidShell off
 AnonRequirePassword off
 <Limit WRITE>
 DenyAll
 </Limit>
 </Anonymous>
 # Default root can be used to put users in a chroot environment.
 # As an example if you have a user foo and you want to put
 # foo in /home/foo
 # chroot environment you would do this:
 # DefaultRoot /home/foo foo
 Defaultroot ~
```

Enregistrez.

Relancez le démon.
Pour ce faire, ouvrez une console et tapez
[root@pc_mic root]# **/etc/rc.d/init.d/proftpd**

La machine m'indique quelles options sont disponibles.
Status : pour savoir si le démon est actif ou non.
Start ou Stop : sans commentaires

Nous allons utiliser ici **restart**
[root@pc_mic root]# **/etc/rc.d/init.d/proftpd restart**

Le démon s'arrête puis redémarre en tenant compte des nouveaux réglages.
N'oubliez pas de faire cette manoeuvre après chaque modification du fichier de configuration.

Les anonymes peuvent maintenant se connecter. Ils ne pourront rien faire tant que vous n'aurez pas mis à leur disposition des fichiers dans le répertoire **/var/ftp/pub**. N'oubliez pas de changer les droits ! Il est rare que root aime voir ses fichiers se balader dans la nature.

vsFTPd : un serveur ftp ultra-sécurisé et simple

Il existe de nombreux serveurs ftp, mais qui sont souvent soit de véritables trous de sécurité, comme l'on en voit quasiment tous les jours sous wu-ftp, soit très gourmands en mémoire comme c'est le cas pour ProFTP.

Mais une solution existe, elle est rapide, sécurisée et fait la joie d'Alan cox, d'IBM et de bien d'autres...

C'est **vsFTPd** (**Very Secure File Transfert Protocol deamon**)

En effet l'auteur, Chris Evans, est parti de la constatation suivante : il n'y a pas de programmeur parfait, et faire un applicatif répondant à toutes les normes de sécurité est difficile, surtout quand cela n'a pas été pensé à la création du projet.

Donc, Chris Evans a pris en compte tous ces problèmes avant même de commencer à coder son application. Et il en est ressorti ceci :

le respect de la gestion des droits et processus en respectant les règles d'Unix
La mise en cage des éléments (cf. chroot) indispensable
la lutte nécessaire contre les "buffer overflows"

Pour cette raison également, de nombreuses commandes utilisées par vsFTPd font partie de l'application et **ne font pas appel au système** (pour le "ls" par exemple). ceci a pour conséquence de sécuriser encore un peu plus le système et d'accélérer le traitement des informations.

Pour cette raison, sa conception est modulaire, chaque partie étant traitée de manière indépendante, réduisant ainsi le nombre d'erreurs de programmation possible, et son optimisation importante.

Pour cette raison aussi, Il est tout à fait possible d'utiliser vsftpd avec xinetd (xinetd se chargeant de lancer le serveur ftp uniquement quand une connexion est demandée)

Exemple de configuration xinetd :

```
service ftp
        {
        socket_type= stream
        wait = no
        user= root
        server= /usr/local/sbin/vsftpd
        server_args =
        log_on_success+= DURATION USERID
        log_on_failure+= USERID
        nice= 10
        disable = no
        }
```

Ceci permet en outre de réguler la bande passante utilisée.

Installation

vsftpd est dans le dépôt Main, le nom du paquet est vsftpd pour une installation vue de Synaptic. Pour installer VsFTPd en console, entrez la commande suivante dans un terminal :

```
sudo apt-get install vsftpd
```

Configuration de vsftpd

La configuration de VsFTPd est centralisée dans un seul et même fichier `/etc/vsftpd.conf`.

Par défaut la configuration standard interdit tout ou presque, c'est à vous et seulement avec quelques lignes de configuration de définir quelles seront les capacités et les accès autorisés ou interdits à l'utilisateur.

Fichiers de configuration :

- /etc/vsftpd/vsftpd.conf,
- /etc/vsftpd.ftpusers,
- /etc/vsftpd.user_list,
- /etc/vsftpd.chroot_list

/etc/vsftpd/vsftpd.conf :
Fichier principal de configuration du service.

/etc/vsftpd.ftpusers :
Liste des utilisateurs qui ne seront pas autorisés à se connecter au service ftp. Tous les autres sont autorisés.

Vsftp par Albert *(www.linuxfrench.net)*

Étudions-la de plus près. Le fichier de configuration fourni dans la distribution source de vsftpd, vsftpd.conf, que vous devez avoir placé dans /etc, est très court une fois toutes les lignes commentées et les explications dissimulées :

```
[~/vsftpd-1.0.1]$ egrep -v '^(#|$)' vsftpd.conf
anonymous_enable=YES
dirmessage_enable=YES
xferlog_enable=YES
connect_from_port_20=YES

- anonymous_enable autorise l'accès anonyme au serveur (comportement par
        défaut) sur le répertoire /var/ftp ou /home/ftp (ubuntu) de
l'utilisateur ftp dans le fichier passwd.
- dirmessage_enable active l'affichage des fichiers .message à l'entrée dans
        les répertoires.
- xferlog_enable active un journal des téléchargements.
- connect_from_port_20 active (bien que ça ne soit pas le comportement par
        défaut du serveur) l'utilisation du port source 20 pour les PORT.

Les trois dernières directives ne sont là que pour outrepasser le comportement
par défaut du serveur, et le rendre un petit peu plus traditionnel, mais il est
amusant de remarquer que vsftpd est parfaitement capable de fonctionner en tant
que serveur FTP à accès anonyme en lecture uniquement, avec un fichier de
configuration entièrement vide.
```

Mon fichier:

```
bureau:/etc # egrep -v '^(#|$)' vsftpd.conf
```

```
listen=YES
anonymous_enable=YES
local_enable=YES #Pour permettre à vos utilisateurs locaux
write_enable=YES #(ceux qui ont un compte sur la machine) de se connecter au serveur
                 #       et d'écrire.
local_umask=077     #Pour permettre la configuration du chmod par défaut
                    # que prendront les fichiers et les dossiers.
dirmessage_enable=YES
xferlog_enable=YES
connect_from_port_20=YES
chroot_local_user=YES #Vous pouvez « emprisonner » certains utilisateurs
                      # dans leur dossier personnel
```

umask:

1. Valeur 002 : permet d'avoir un chmod des dossiers en 775 et fichiers en 664
2. Valeur 022 : permet d'avoir un chmod des dossiers en 755 et fichiers en 644
3. Valeur 077 : permet d'avoir un chmod des dossiers en 777 et fichiers en 666

- Vous pouvez personnaliser le texte de connexion au serveur

```
ftpd_banner=Bienvenue sur le serveur ftp de jojo
```

- Vous pouvez « emprisonner » certains utilisateurs dans leur dossier personnel grâce à la configuration suivante :

```
chroot_local_user=YES
chroot_list_enable=YES

chroot_list_file=/etc/vsftpd.chroot_list
```

Les identifiants des utilisateurs concernés doivent êtres renseignés dans */etc/vsftpd.chroot_list*.

- Pour avoir un monitoring basic (voir les utilisateurs connectés) :

```
setproctitle_enable=YES
```

Ensuite, pour voir la liste des utilisateurs il suffit de taper la commande suivante dans un terminal :

```
ps -aef | grep vsftpd
```

Attention, tous les utilisateurs du système ayant accès à la commande "ps" pourront également visualiser la liste des utilisateurs connectés.

* Monitoring d'après le fichier de log

Aucun changement à effectuer dans vsftp.conf. Pas de problème de sécurité car la commande est accessible en mode super-utilisateur seulement. Tapez dans un terminal :

```
sudo tail -f /var/log/vsftpd.log
```

Après chaque changement sur la configuration, pensez à relancer le serveur pour les prendre en compte :

```
sudo /etc/init.d/vsftpd restart
```

Un autre fichier "standard" de configuration de vsFTPd :

anonymous_enable=NO	connexion en tant qu'anonyme interdite
local_enable=YES	connexion pour les utilisateurs locaux autorisés
write_enable=NO	écriture interdite
anon_upload_enable=NO	Upoad pour anonyme interdit
anon_mkdir_write_enable=NO	création de répertoire pour les anonymes interdit
anon_other_write_enable=NO	écriture de fichier pour anonyme interdit
chroot_local_user=YES	enferme l'utilisateur dans le répertoire ftp (chrooté)
per_source = 5	nombre de connexions maximales autorisées par une même adresse IP
no_access = 192.168.1.3	interdiction de se connecter au ftp à partir de cette adresse
guest_enable=YES	Rend possible la connexion d'utilisateurs virtuels
guest_username=virtual	souhaitable, vu que la connexion anonyme est refusée
pasv_min_port=30000	cette option permet de limiter l'accès des ports
pasv_max_port=30999	disponibles, utile derrière un firewall
xferlog_enable=YES	enregistrement des logs...

simple et court

Mais les options sont bien plus nombreuses!.

Ouvrir le port FTP en utilisant iptables (optionnel)

Ajouter les règles suivantes à votre script iptables . Si on autorise les entrées:

Si le protocol est TCP et le port est 21 (ftp):

```
iptables -A INPUT -p tcp -m tcp --dport 21 -j ACCEPT
```